# Report quality (10)

## Clearly stating project objectives

## inpaint the picture with missing area, try to make it contextually consistent

## gain some understandings on GAN

## catch up with cutting edge technologies regarding deep learning

## Clearly stating prior work and what was used as a basis

## talking about the architecture in the paper \*

## mention another brief model/paper description (if necessarily)

## training set used (pictures etc.)

## modified model (modifications made)

1. Clearly stating accomplishments (see quality above)

## training processes (compare the previous one, approaches we tried including the global)

## result before and after revision (put some good ones) and compare

1. Clearly describing future work (if you were to continue on this or if another team picked up from where you left off)
2. Proper references (included any figures you did not create)

Report Outline

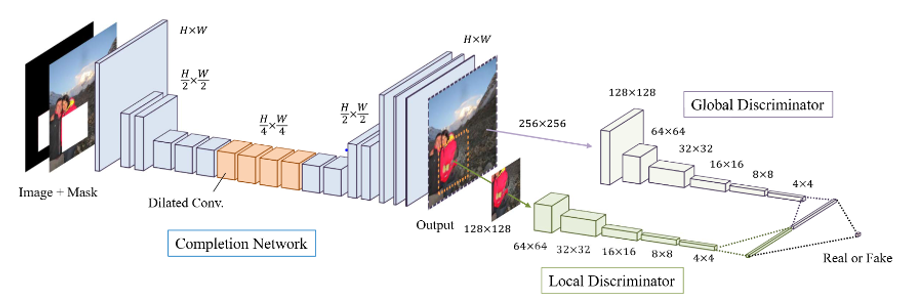
1. Project objectives
2. Prior work and model architecture
3. Dataset collection
4. Modifications
5. Accomplishments
6. Future works
7. **Project objectives**

The objective of our project is to build a GAN which can inpaint landscape images with a missing area. The landscape picture will be covered by a square blank mask and our model will recover the original picture by making the missing area as real and natural as possible, as well as contextually consistent with the rest of the picture. We were able to accomplish this by taking as reference an existing model that does this task in [1] and by proposing and evaluating modifications to that model in order to get the results we desire.

In addition, through our research on the existing GAN models and our work on building a GAN, our objective is to gain more understanding as well as experience on GANs. GAN being a relatively new and exciting topic in deep learning, we were able to further build on the knowledge on GANs presented in class and learn first hand all the challenges that come with training a GAN. Furthermore, we got familiar with cutting edge technologies and applications regarding deep learning while in the process of choosing our project topic.

1. **Prior work and model architecture**

In paper [1], it mentioned a model that worked well in inpainting. The author involved training two networks which are completion network (generator) and discriminator. As it is shown below, the model for the paper is the one that we are referring.



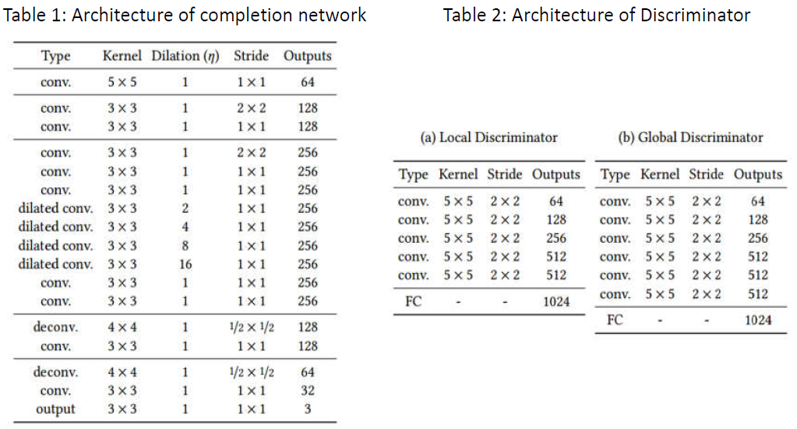
As the figure shows, the original image will be applied with the mask and later the 3 channels RGB image with missing part will be taken along with the mask as the whole input to the generator network. Briefly, the input image data will be processed by convolutional layers in CNN to be resized to quarter of the original size. Specifically, the data will go through 4 dilated convolutional layers which help the network gain a better receptive field of the picture as a whole. In other words, the kernel is applying on more pixels far away from center to enable the network to see more. Lastly the completion network will restore the size of the processed data.

For discriminator part, the original model contains two discriminators which are global discriminator and local discriminator. They are considered together to improve the contextual consistency of the generated pictures. The filled part of the image and the full-size generated image will be the input to local and global discriminator respectively. In the end for classification, the two part will be resized, flatten and then combined as the input of the fully connected network.

According to the description, the table given below shows more details of the original model.

Based on the model mentioned above, we made some changes to make the new model fit to the project. Basically, the generator setting is the same according to layers design but MSE loss is instead used for generator training.

And for discriminator, rather than considering two we only keep the local discriminator. Moreover, instead of doing classification we use the real and fake partial image to calculate the Wasserstein distance as the loss function. The goal of the process is to try to fit the real context of the image.



1. **Dataset collection:**

In this part, we downloaded 145K photos from Flickr with commercial use allowed license. The original photos size is not smaller than 350px in height or width. Our generator is a pixel2pixel architecture so that it is compatible to all resolution input images, but we crop these images into 256x256 resolution for the convenience of batch training.

Due to the size of dataset is quite large for this one month project. In the training procedure, we only chose 96000 images as our training dataset, and all experiment processes mentioned below is on that small dataset.

|  |  |
| --- | --- |
| 图片包含 屏幕截图  描述已自动生成 |  |

1. **Modifications and Problem occurred**
2. Modification of loss

The original paper we referred to was using DCGAN to train their deep networks. However, we read some papers mentioned that DCGAN is hard to train. It requires many hyper-parameters adjustment during the training process and also needs the distributions of generated images and real images have common crossed areas. That means their common distributions measurement should be bigger than 0 which is quite hard to satisfied. Also, DCGAN loss cannot represent the process of training. Therefore, we turned to WGAN-GP to avoid these problems.

Different from DCGAN loss for Discriminator which is trying to separate generated images and real images, WGAN-GP loss defines a measurement named Wasserstein distance to represent the distribution distance between generated images and real images and tries to minimize it. Therefore, the discriminator loss of WGAN-GP represents the similarity of generated images and real images.

1. Modification of training process

When training the network, we firstly trained discriminator 3 times and generator once, and find that the loss of discriminator diverged quickly. Then we changed the strategy, we trained discriminator once and followed with 10 times generator training. The loss went well at the beginning, but unstable after few minutes. We noticed that this architecture may not compatible with momentum optimizer, so we changed optimizer to Rmsprop finally and it went well. However, the training process was quite time consuming and it cost days but achieved few result. The original paper mentioned that they use 8 GPUs and trained 2 months to get a roughly satisfied result.

Due to the limited project time and resource we can use. We then changed a little about the training process, we firstly use MSE loss to train the generator trying to make it be able to generate some roughly pieces of contents, and then use WGAN-GP to enhance the results.

1. Modification of convolution padding

In out implemented code, we did not use the zeros padding method to pad every convolutional layer. We used ‘REFLECT’ padding to replace the default padding strategy. We think reflect padding method can keep information as much as possible for dilate convolutional operation.

1. **Accomplishments**
2. MSE pre-training result

We use MSE loss to pretrain the generator for about one day and get result like below:

|  |  |  |
| --- | --- | --- |
| 图片包含 树  描述已自动生成 | 图片包含 水, 户外, 天空, 男士  描述已自动生成 | 图片包含 自然, 户外, 摇滚  描述已自动生成 |
| 图片包含 户外, 海滩, 鸟, 水  描述已自动生成 |  | 图片包含 户外, 天空, 自然, 日落  描述已自动生成 |
| 图片包含 水, 海滩, 天空, 户外  描述已自动生成 | 图片包含 水, 户外, 自然  描述已自动生成 | 图片包含 水, 户外, 山, 自然  描述已自动生成 |

1. WGAN-GP training result

|  |  |  |  |
| --- | --- | --- | --- |
| Generated images | Original images | Generated images | Original images |
| 图片包含 户外, 天空, 山, 摇滚  描述已自动生成 | 图片包含 户外, 天空, 山, 自然  描述已自动生成 | 图片包含 水, 户外, 天空, 场景  描述已自动生成 | 图片包含 户外, 水, 天空, 自然  描述已自动生成 |
| 图片包含 摇滚, 植物, 自然  描述已自动生成 | 图片包含 植物, 摇滚  描述已自动生成 | 图片包含 天空, 树, 户外, 日落  描述已自动生成 | 图片包含 天空, 树, 户外, 水  描述已自动生成 |
| 图片包含 户外, 树, 汽车, 卡车  描述已自动生成 | 图片包含 户外, 树, 汽车, 草  描述已自动生成 | 图片包含 水, 户外, 天空, 自然  描述已自动生成 | 图片包含 水, 天空, 户外, 自然  描述已自动生成 |
| 图片包含 水  描述已自动生成 | 图片包含 水, 自然  描述已自动生成 | 图片包含 自然, 水, 瀑布, 户外  描述已自动生成 | 图片包含 自然, 水, 瀑布, 户外  描述已自动生成 |
| 图片包含 山, 户外, 雪花, 天空  描述已自动生成 | 图片包含 户外, 山, 天空, 树  描述已自动生成 | 图片包含 草, 天空, 户外, 山  描述已自动生成 | 图片包含 草, 天空, 户外, 山  描述已自动生成 |
| 图片包含 天空, 户外, 水, 场景  描述已自动生成 | 图片包含 户外, 天空, 黄色, 水  描述已自动生成 | 图片包含 山, 天空, 户外, 水  描述已自动生成 | 图片包含 山, 天空, 户外, 雪花  描述已自动生成 |

Although MSE pretrained generator only captured roughly contents in original pictures, but it makes generator training much faster in WGAN training process than before. However, the training process based on WGAN is also becoming much slower after around 50 epochs.

1. **Future works**
2. Due to that we want to simplify the training process and shorten our training time to get some results before project deadline, we fixed the missing mask position in the final modified method. But the ideal situation is that the mask can be anywhere in the input images. Therefore, the first future works should be that randomly choose mask position and mask size to train the network.
3. Due to the input of the generator is an image with blank area in mask region and the generator improves much slower in WGAN-loss optimizer, it is quite hard to generate the best result directly. An alterative way is that split generator into two parts, one is what we have done here to generate a result which is roughly fine but lacks content in detail. The other part is a refine part, use the first stage result to get a better one. The stage 1 result is different from the very beginning’s input, it does have content information about mask area, this information is important to guide the stage 2 process to refine the missing parts of image.
4. **Reference**

(add here)